MPKC, Toronto

june 1999

Speeding up the discrete log computations on curves with automorphisms

joint work of I. Duursma (Univ. Limoges), P. Gaudry, F. Morain (cole polytechnique, LIX)

duursma@unilim.fr,

gaudry,morain@lix.polytechnique.fr

Plan

I. The problem.

II. Generic methods.

III. Parallel collision search with an automorphism.

IV. Gaudry's variant of the Adleman-DeMarrais-Huang algorithm.

I. The problem

$\mathbb{K} = \mathbb{F}_q = \mathbb{F}_{p^r}$

$C/\mathbb{K}$ smooth projective curve

Input: $P \in \mathrm{Jac}(C)$ of order n, $Q \in < P >$.

Output: k s.t. $Q = [k]P$.

Ex. elliptic case

$C/\mathbb{K} : Y^2 + a_1 XY + a_3 Y = X^3 + a_2 X^2 + a_4 X + a_6$

or hyperelliptic case:

$$C/\mathbb{K} : Y^2 + H(X)Y = X^{2g+1} + \cdots$$

Benchmark CERTICOM exercises (cf. `http://www.certicom.com/chal`).

The discrete log problem on elliptic curves

- No subexponential method in general.
- Exceptions for:
  - ˘ supersinguler curves: MOV (Menezes, Okamoto, Vanstone);
  - ˘ curves of trace 2: Tate's pairing (Frey & Rck);
  - ˘ curves of trace 1: log on $\mathbb{Q}_p$ (Satoh-Araki; Semaev; Smart).
  - ˘ XEDNI: Silverman *et al.*, Cheon *et al.*???

Discrete log on hyperelliptic curves

- Some particular cases *la* Rck.
- Algorithm ADH from Adleman, DeMarrais, Huang (ANTS I):

$$L_{p^{2g+1}}[1/2, c]$$

with $c \leq 2.181$ if $\log p \leq (2g+1)^{0.98}$ ($\leftarrow$ Lovorn's theorem on smooth polynomials); SNF.

- Extensions, precise analysis and optimizations by Enge:

$$L_{q^g}[1/2, c(\theta)]$$

if $\theta \log q \leq g$, with $\lim_{\theta \to 0} c(\theta) = +\infty$; easier SNF.

- Flassenberg & Paulus: using sieving techniques; experiments with $y^2 = x^{2g+1} + 2x + 1$, faster than Shanks for $g \geq 6$.
- PG's new variant: fast, practical, implemented, breaks a lot of proposed systems.

II. Generic methods

A) Shanks

$$k \in [-(n-1)/2, (n-1)/2], \quad w = \sqrt{(n-1)/2}$$

$$k = uw + v, 0 \leq v < w, \quad |u| \leq w.$$

$$[k]P = Q \iff [u]([w]P) = Q \ominus [v]P.$$

baby steps: compute the points $Q \ominus [v]P$ for all v's in $O(w)$ steps.

giant steps: find u in $O(w)$ steps.

Cost: $O(w) = O(\sqrt{n})$ in time and space.

B) Pohlig-Hellman

$$n = \prod_i p_i^{\alpha_i}$$

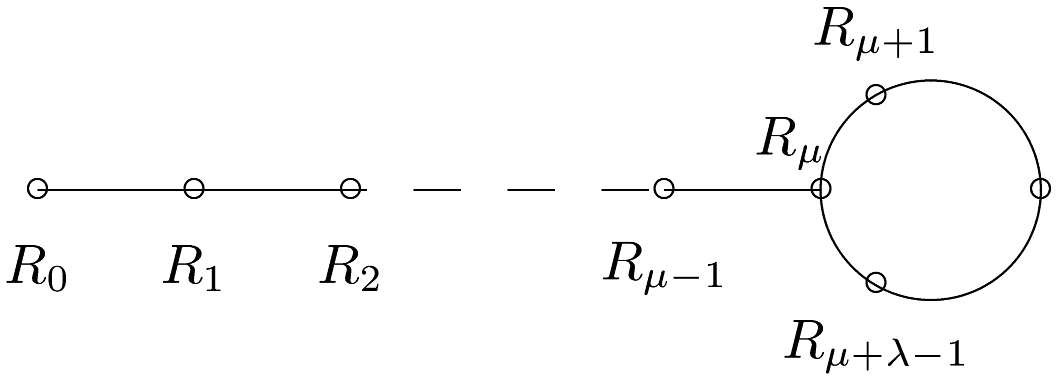
Idea: finding $[x]P = Q$ is equivalent to finding $x \bmod n$, i.e., $x \bmod p_i^{\alpha_i}$ for all i (CRT). It reduces the discrete log $\bmod n$ to that $\bmod p_i^{\alpha_i}$ for all i.

Cost: $O(\max(\alpha_i \mathrm{DL}(p_i)))$.

C) Pollard's ρ

Idea: use the properties of functional graphs.

Let f be a random function: $< P > \to < P >$,
$R_0 \in < P >_R$ and build $R_{i+1} = f(R_i)$:



More precisely, one starts from $R_0 = [u_0]P + [v_0]Q$ with u_0 and v_0 in $\mathbb{Z}/n\mathbb{Z}_R$ and one builds the sequence $R_{i+1} = f(R_i)$, which also builds (u_i) and (v_i) modulo n s.t. $R_i = [u_i]P + [v_i]Q$.

Random mapping statistics; (Flajolet & Odlyzko, EUROCRYPT'89) a functional graph for a random f with n vertices should have $0.5 \log n$ components, $\overline{\mu}_n = \overline{\lambda}_n = \sqrt{\pi n/8}, \overline{\rho}_n = \sqrt{\pi n/2}$.

Clon: after $\approx \sqrt{\pi n/2}$ iterations, we should find a collision, that is i and j s.t. $R_i = R_j$:

$$[u_i - u_j]P + [v_i - v_j]Q = 0$$

hence k if $v_i \neq v_j$ (which happens with probability $1/n$).

In practice: there is an $e = O(\sqrt{n})$ s.t. $R_e = R_{2e}$:

- $R := R_0$, $S := R_0$;
- **repeat** $R := f(R)$, $S := f(f(S))$ **until** $R = S$.

Cost: time $O(\sqrt{n})$ and space $O(1)$.

IV. Parallel collision search with an automorphism

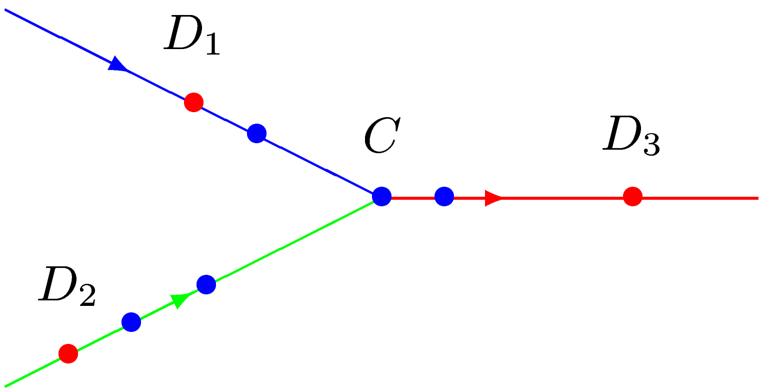
A) Distinguished points

(Quisquater & Delescaille, EUROCRYPT'89)

Idea: store points with a distinguishing property (e.g., $P = (x, y)$ and $x \equiv 0 \bmod 2^b$). If θ is the proportion of distinguished points, one should find a collision after having computed $\theta\sqrt{\pi n/2}$ distinguished points.

B) Parallel collision search

(van Oorschot & Wiener, *J. Cryptology*, 1999)



M processors run through the graph at the same time: one should find a collision after each processor has computed $\sqrt{\pi n/2}/M$ points, or $\theta\sqrt{\pi n/2}/M$ distinguished points.

Rem. All solved exercises of Certicom done this way.

Choosing f

- additive random walk:
  - precompute r points $(T^{(j)})_{0\leq j<r}$ in $< P >$, e.g. $T^{(j)} = [u^{(j)}]P + [v^{(j)}]Q$.
  - $f(R) = R + T^{(j)}$, with $j = \mathcal{H}(R)$ where $\mathcal{H}$ sends $< P >$ to $\{0, 1, \ldots, r-1\}$. In other words: $R_i = [u_i]P + [v_i]Q$, with $u_{i+1} \equiv u_i + u^{(j)} \bmod n, \quad v_{i+1} \equiv v_i + v^{(j)} \bmod n$.
  - Satler & Schnorr: $r \geq 8$; Teske: $r \geq 20$ (+ Hildebrand's result on random walks $\bmod n$).

- multiplicative random walk: (Gallant *et al.*)
  - choose multipliers $(\mu_j)_{0\leq j<r}$;
  - $f(R) = [\mu_j]R$, $j = \mathcal{H}(R)$, i.e.:

$$f(R) = \left[\prod_{j=0}^{r-1} \mu_j^{f_j}\right] R_0.$$

  - requires at least two processors!
  - computing $[\mu_j]R$ must be done quickly.

Using equivalence classes

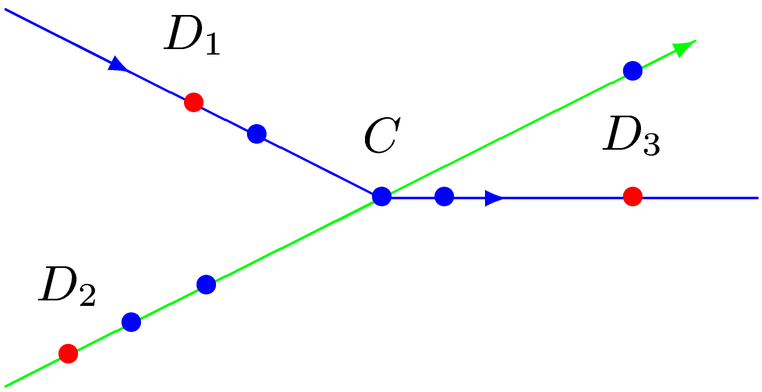
Idea: let $\sim$ be an equivalence relation on $< P >$ s.t. $\#(< P > / \sim) = n/m$. If we can do a random walk on this set, we should find a collision after $\sqrt{\pi n/(2m)}$ iterations.

Finding $\sim$: use an automorphism α of order m of C:

$$S \sim T \iff \exists i, S = \alpha^i T$$

Ex. (Wiener, Zuccherato) $(x, y) \sim (x, -y)$.

1st idea: iterate $R_{i+1} = f(R_i)$ and find $\overline{R_i} = \overline{R_j}$.



Ex. $R_{i+1} = R_i + T^{(k)} = (x, y)$,
$S_{j+1} = S_j + T^{(\ell)} = (x, -y)$, $\overline{S_{j+1}} = \overline{R_{i+1}}$, but in general $f(S_{j+1}) \neq f(R_{i+1})$.

$\Rightarrow$ well defined function: $R' \in \overline{R} \Rightarrow \overline{f(R')} = \overline{f(R)}$.

2nd idea: $R_{i+1} = \overline{f(R_i)}$.

Short cycles

Pb: suppose we define $\overline{R}$ to be the least point in lexicographical order in $x = R_x$.

$$R_{i+1} = \overline{R_i + T^{(j)}} = -R_i - T^{(j)}.$$

We can have:

$$R_{i+2} = \overline{R_{i+1} + T^{(j)}}$$

(the same j)

$$R_{i+2} = \overline{-R_i - T^{(j)} + T^{(j)}} = R_i.$$

More generally (with $\varepsilon_k = \alpha^{e_k}$):

$$\begin{tikzcd}
R_1 \arrow[d, "j_1"', "\varepsilon_1"] & R_t \arrow[l, "j_t"', "\varepsilon_t"] \\
R_2 \arrow[r, dotted, bend right] & R_{t-1} \arrow[u, "j_{t-1}", "\varepsilon_{t-1}"']
\end{tikzcd}$$

Thm. the probability $\mathcal{P}(t)$ to have a trivial t-cycle is:

t	pattern	$\mathcal{P}(t)$	
2	$j_1 = j_2$	$1/(mr)$	
3		$\leq 2/(mr)^2$	$3 \mid m$
		0	$3 \nmid m$
4	$j_1 = j_2 = j_3 = j_4$	$\leq (1-1/m)^2/(mr^3)$	
	$j_1 = j_3, j_2 = j_4$	$(1-1/r)/(mr)^2$	

Idea of proof:

$$R_{k+1} = \varepsilon_k(R_k + T^{(j_k)})$$

$$R_{t+1}/\varepsilon_t = \varepsilon_{t-1} \cdots \varepsilon_1 R_1 + \tau$$

with $\tau = \varepsilon_{t-1} \cdots \varepsilon_1 T^{(j_1)} + \varepsilon_{t-1} \cdots \varepsilon_2 T^{(j_2)} + \cdots + \varepsilon_{t-1} T^{(j_{t-1})} + T^{(j_t)}$. We have a trivial cycle if $\tau = 0$.
$\square$

Collapsing the cycle: replace the input point by a symmetrical expression in the cycle points, e.g. $R = \sum_{i=1}^{t}[i^i + 1]S_i$.

1st Example: the ABC curves

Koblitz (CRYPTO '91): $\mathbb{K} = \mathbb{F}_{2^\ell}$ and E is defined over $\mathbb{F}_2$. Then $\lambda : (x, y) \mapsto (x^2, y^2)$ is an automorphism of order ℓ.

Gallant, Lambert et Vanstone use $f(R) = [\mu_{\overline{R}}]R$, which is well defined since

$$f(R') = [\mu_{\overline{R'}}]R' = [\mu_{\overline{R}}]R' = [\mu_{\overline{R}}]([\pm\lambda^i]R)$$
$$= [\pm\lambda^i]([\mu_{\overline{R}}]R) \in \overline{f(R)}.$$

Idea: take $\mu_{\overline{R}} = 1 + \lambda^{\mathcal{H}(\overline{R})}$.

Rem. The (μ_i) should be multiplicatively independant.

Theoretical results on automorphisms

Rem. $\mathrm{Aut}(\mathcal{C}) \hookrightarrow \mathrm{Aut}(\mathrm{Jac}(\mathcal{C}))$.

Thm. (Roquette 1970) If $g > 1$ and $p > g + 1$,

$$\#\mathrm{Aut}(\mathcal{C}) \leq 84(g-1).$$

(exception: $y^2 = x^p - x$)

Thm. (Stichtenoth 1973) If $g > 1$ with no restriction on p,

$$\#\mathrm{Aut}(\mathcal{C}) \leq 16g^4.$$

(exception: $y^q - y = x^{q+1}$)

Rem.

- too many automorphisms $\Longrightarrow$ split jacobian
- too many automorphisms $\Longrightarrow$ supersingular curve

New examples

A) elliptic curves

Only two cases with non-trivial automorphisms:

E	autom.	order
$Y^2 = X^3 + aX$	$[i](x, y) = (-x, iy)$	4
$Y^2 = X^3 + b$	$[-\rho](x, y) = (\rho x, -y)$	6

with $i^2 = -1$, $\rho^3 = 1$.

Rem. can be combined with the Frobenius when applicable.

Rem. multiplicative random walks cannot be used, since:

$$\alpha^2 = -1, \quad (1+\alpha)(1-\alpha) = 2.$$

B) Hyperelliptic curves

$$Y^2 + H(X)Y = F(X) = X^{2g+1} + \cdots$$

Group law: defined on $\mathrm{Jac}(C)$, a group of size $\simeq q^g$.
Addition done using Cantor's algorithm.

Representing divisors:

If $D \in \mathrm{Jac}(\mathcal{C})$, then
$D = P_1 + P_2 + \cdots + P_g - g \cdot (\infty)$. There is a unique representation as pairs of polynomials:

$$D = [u(z), v(z)],\ \deg(u) \leq g,\ \deg(v) < \deg(u),$$

defined by (if $P_i = (x_i, y_i)$),

$$u(z) = \prod_{i=1}^{g}(z - x_i),\ \text{and } v(x_i) = y_i,\ \forall i.$$

Rem. If $D = [u(z), v(z)]$, then $-D = [u(z), -v(z)]$.

Rem. Cantor's algorithm uses this representation.

Action of the automorphism

If α is an automorphism of C:

$$\alpha(x, y) = (\alpha_1(x, y), \alpha_2(x, y)).$$

We can deduce:

$$\tilde{\alpha} \ : \ \mathrm{Jac}(\mathcal{C}) \longrightarrow \mathrm{Jac}(\mathcal{C}).$$

Rem. Can be combined with the hyperelliptic involution (speedup of $\sqrt{2}$).

Example: $Y^2 = X^5 - 1$

$$\alpha(x, y) = (\zeta_5 x, y), \quad \text{where } \zeta_5^5 = 1.$$

$\Rightarrow -\alpha$ has order 10.

Action on divisors:

$$[z^2 + u_1 z + u_0, v_1 z + v_0] \mapsto [z^2 + \zeta_5 u_1 z + \zeta_5^2 u_0, \\ -\zeta_5^{-1} v_1 z - v_0].$$

Rem. The multiplicative walk is not possible.

IV. Gaudry's variant of the Adleman-DeMarrais-Huang algorithm

or how to make NS paranoid at last!

Main features:

- easy to describe and to analyze.
- index calculus using a basis of S-smooth divisors, i.e. $G = \{[u(z), v(z)] \in \mathrm{Jac}(C), u(z)$ is irreducible of degree $\leq S\}$, with $S = 1$ for $g \leq 8$.
- Finding relations using a random walk.
- Easier linear algebra.
- exponential asymptotic complexity but very practical and breaks a lot of systems proposed by various authors.

The algorithm

Input: $D_1, D_2 \in \mathrm{Jac}(C), n = \mathrm{ord}(D_1)$ known.

Output: λ s.t. $D_2 = [\lambda]D_1$.

1. [Initialisations:] Fix a smoothness bound S and build the basis $G = \{g_1, g_2, \ldots, g_w\}$. Compute $T^{(j)} := [\alpha^{(j)}]D_1 + [\beta^{(j)}]D_2$ for $j \in [1..r]$ and set $R_0 := [\alpha_0]D_1 + [\beta_0]D_2$. Put $k := 1$.

2. [Main loop:]

   (a) Look for a smooth divisor:

   repeat

   $$j := \mathcal{H}(R_0), R_0 := R_0 + T^{(j)},$$

   $$\alpha_0 + = \alpha^{(j)} \bmod n, \beta_0 + = \beta^{(j)} \bmod n.$$

   until $R_k := R_0 = [u_0(z), v_0(z)]$ is S-smooth.

   (b) [Write R_0 on G:] factor $u_0(z)$ over $\mathbb{F}_q$, store $R_k = \sum m_{ik} g_i$ and $\alpha_k = \alpha_0, \beta_k = \beta_0$.

   (c) If $k < \#G + 1$, then $k := k + 1$, go to 4.

3. [Linear algebra:] find (γ_k) in $\mathrm{Ker}({}^t M)$ on $\mathbb{Z}/n\mathbb{Z}$.

4. Return $\lambda = -(\sum \alpha_k \gamma_k)/(\sum \beta_k \gamma_k) \bmod n$.

Analysis for $S = 1$

(enough for $g \leq 8$)

Prop. The probability for $u(z) \in \mathbb{F}_q[z]$ of degree g to be 1-smooth is $1/g!$.

- building G: $\#G = O(q) \Rightarrow O(q)$ computations of squareroots in $\mathbb{F}_q$.
- random walk: $O(g!q)$ additions on the jacobian and smoothness test of a degree g polynomial, followed by $O(q)$ EDF's of degree g polynomials (total $\leq O(gq)$).
- linear algebra: with Lanczos on a $q \times q$ matrix over $\mathbb{Z}/n\mathbb{Z}$, each row having $\leq g$ entries $\Rightarrow$ $O(q(qg)(\log n)^2) = O(g^3 q^2 (\log q)^2)$ (since $n \approx q^g$).

Thm. Gaudry's variant takes time $O(q^2 \log^c q)$ for fixed g.

Fine points: smoothness test (use discriminant *la* Swan), EDF, SGE for fast linear algebra ($\neq$ factorization, $\neq$ discrete log over $\mathbb{F}_{p^n}$).

More properties

Rem. The algorithm has asymptotic exponential complexity, but works very well in practice (see below), since it must factor polynomials of smaller degree than ADH. Provably (and practically) better than Pollard's ρ for $g > 4$.

Size of Jac breakable: $\#\mathrm{Jac} = q^g = (2\#G)^g$:

$\#G \backslash g$	4	5	6	7	8
10^5	2^{70}	2^{88}	2^{105}	2^{123}	2^{140}
10^6	2^{83}	2^{104}	2^{125}	2^{146}	2^{167}

Crucial rem. An automorphism of order m reduces $\#G$ by a factor m and the linear algebra by m^2.

Hyperelliptic cryptosystems

Author	equation	q	auto
Koblitz	$Y^2 + Y = X^5 + X^3 \ (*)$	2^ℓ	4ℓ
	$Y^2 + Y = X^5 + X^3 + X \ (*)$	2^ℓ	2ℓ
	$Y^2 + Y = X^{2g+1} + X$	2^ℓ	2ℓ
	$Y^2 + Y = X^{2g+1}$	2^ℓ	2ℓ
Spallek	$Y^2 = X^5 - 140X^3 - 240X^2 + 3810X + 6928$	p	2
Buhler Koblitz Chao et al.	$Y^2 + Y = X^{2g+1}$ (+ twists)	p with $p \equiv 1 \ (2g+1)$	$2(2g+1)$
Sakai Sakurai Smart	$Y^2 + Y = X^{13} + X^{11} + X^9 + X^5 + 1$	2^{29}	4×29
Duursma Sakurai	$Y^2 = X^p - X + 1$	p^ℓ	$2\ell p$

Timings

Magma 2.3 on Pentium II 450 MHz, 128 Mb.

Example 1:

field	$\mathbb{F}_{2^{19}}$
equation	$y^2 + (x+1)y = x^9 + x^7 + x^6 + x^5 + x^4 + x^3 + 1$
genus	4
auto	2×19
$\#\mathrm{Jac}$	$2 \times 3 \times 12628657530494945987191 \approx 2^{76}$
basis	$13837 \ (262885)$
matrix	30000×13837

construct. basis	145 s
relations	2 h
exp. # rel.	23.88
SGE	14 h
Lanczos	2 h
Total	18 h

Rem. non optimized SGE.

Example 2:

field	$\mathbb{F}_{2^{19}}$
equation	$y^2 + (x+1)y = x^{11} + x^9 + x^8 + x^7 + x^4 + x + 1$
genus	5
auto	2×19
$\#\mathrm{Jac}$	$2^5 \times 1236507505106065358922109709 \approx 2^{95}$
basis	$13782\ \ (261840)$
matrix	30000×13782

construct. basis	145 s
relations	3.5 h
exp. # rel.	119.7
SGE	15.5 h
Lanczos	7 h
Total	26 h

Example 3:

field	$\mathbb{F}_{2^{19}}$
equation	$y^2 + (x+1)y = x^{13} + x^{12} + x^{11} + x^{10}$ $+x^8 + x^6 + x^5 + x^4 + x^3 + x^2 + 1$
genus	6
auto	2×19
$\#\mathrm{Jac}$	$2 \times 23 \times 45660353558951618050075673976938 3 \approx 2^{114}$
basis	$13953 \;\; (265089)$
matrix	30000×13953

construct. basis	145 s
relations	9 h
exp. # rel.	686.9
SGE	16 h
Lanczos	17 h
Total	42 h

Example 4: (Koblitz)

field	$\mathbb{F}_{133349}$
equation	$y^2 + y = x^{17}$
genus	8
auto	2×17
$\#\mathrm{Jac}$	$17^3 \times 2039718158933279430874232765878334605 1 \approx 2^{136}$
basis	$3932 \ (66828)$
matrix	4012×3932

construct. basis	50 s
relations	8.5 d
SGE	30 m
Lanczos	5 h
Total	9 d

Perspectives

Generalization: discrete log in $\mathbb{F}_{q^g}^*$ when q is small.

Limiting factor: linear algebra stage, not the collection of relations ($\approx$ factorization).

More curves to break:

- Koblitz-Buhler: $y^2 + y = x^{13}$ (genus 6) on $\mathbb{F}_p$ has an automorphism of order 2×13:
  - being done: $p = 1,007,059$, $\#\mathrm{Jac} \approx 2^{120}$, $\#G = 38,960$.
  - future: $p \approx 5,000,000$, $\#\mathrm{Jac} \approx 2^{134}$, $\#G = 185,000$.
- Sakai Sakurai Smart: $y^2 + y = x^{13} + x^{11} + x^9 + x^5 + 1$ (genus 6) on $\mathbb{F}_{2^{29}}$ with an automorphism of order 4×29, $\#\mathrm{Jac} \approx 2^{174}$. Requires $\#G \approx 4,500,000$.

Conclusions

- Be very careful when using curves with an automorphism of large order.
- Find curves with cardinality easy to compute but without automorphisms: use random curves or maybe $X_0(N)$ (which has only involutions).
- Low genus is 5 or maybe 4 or 3 (surely not 1)? Implies make the base field larger (bad news for smart cards).
- buy random elliptic curves!